

VŠB - Technická univerzita Ostrava
Fakulta elektrotechniky a informatiky

BAKALÁŘSKÁ PRÁCE

2010

Petr Vaniš

VŠB - Technická univerzita Ostrava
Fakulta elektrotechniky a informatiky

Katedra informatiky

Portál vojensko-geografických informací
The Portal of Military and Geographical
Informations

2010

Petr Vaniš

Prohlášení

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně. Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

V Praze dne 3. května 2010

Petr Vaniš

Prohlášení zástupce spolupracující právnické osoby

Souhlasím se zveřejněním této bakalářské práce dle požadavků čl. 26, odst. 9 Studijního a zkušebního řádu pro studium v bakalářských/magisterských programech VŠB-TU Ostrava.

V Praze dne

mjr. Ing. Otakar Růžička

Poděkování

Na tomto místě bych rád poděkoval své vedoucí bakalářské práce, paní Daniele Ďurákové, za cenné rady a množství času věnované konzultacím. Dále bych rád poděkoval, panu Otakaru Růžičkovi, za uvedení do problematiky mapových serverů a za neomezenou podporu během zpracovávání této bakalářské práce.

Abstrakt

Tato bakalářská práce se věnuje problematice informačního systému sloužícího potřebám vojenských geografů. Účelem systému je zpřístupnit geografické produkty pomocí webových technologií a umožnit sdílení těchto produktů v intranetu armády ČR. Systém má za úkol sjednotit způsob přístupu k různým typům geografických produktů získaným z různých zdrojů. Dále systém řeší sdílení produktů vytvořených jeho uživateli a přístup k datům podle přidělených oprávnění. Tato práce popisuje vznik systému od definice požadavků až po implementaci první verze.

Klíčová slova

Mapa, informační systém, geograf, MapServer, PHP.

Abstract

This thesis deals with the issue of information system which serves the military geographers purposes. The purpose of the system is to make geographical products available with use of web technology and to allow sharing of these products in the intranet of the army of Czech Republic. The aim of the system is to unify the way of access to different types of products from different sources. In addition the system solves sharing of products created by its users and access to data according to privileges granted by system administrator. This work describes the genesis of the system from a definition of requirements to implementation of the first version of the system.

Key Words

Map, information system, geographer, MapServer, PHP.

Obsah

1	Úvod.....	4
2	Kde se vzala potřeba portálu?.....	5
2.1	Potřeby vojenských geografů.....	5
2.2	Proč nevyhovují zavedené firemní systémy?.....	6
2.3	Analýza stavu.....	7
3	Výběr technologií.....	8
3.1	MapServer.....	8
3.2	PostgreSQL.....	9
3.3	PHP5.....	9
3.4	GNU/Linux a Apache.....	9
3.5	Javascript a AJAX.....	10
3.6	Co a proč jsem nepoužil.....	10
3.6.1	OpenLayers.....	10
3.6.2	jQuery.....	10
3.6.3	Další nepoužité technologie.....	11
4	Analýza systému.....	12
4.1	Požadavky.....	12
4.1.1	Hlavní představa.....	12
4.1.2	Funkční požadavky.....	12
4.1.3	Nefunkční požadavky.....	14
4.2	Datová analýza.....	15
4.2.1	GEMIS.....	15
4.2.2	Internet.....	16
4.2.3	PGZ.....	16
4.3	Případy užití.....	21
4.3.1	User.....	22
4.3.2	Geographer.....	22
4.3.3	Supervisor.....	23
4.4	Diagramy aktivit.....	23
4.4.1	Procházení produktů.....	23
4.4.2	Přidání nového produktu.....	25
4.5	Třídní diagram.....	25
4.5.1	Generalizace aktéra.....	26
4.5.2	Rozhraní.....	27
4.5.3	Kolekce.....	27
5	Implementace.....	28
5.1	Masky přístupu.....	28
5.2	Počítání souřadnic.....	29
5.3	Zobrazení pokrytí.....	31
6	Nasazení.....	34
6.1	Topologie.....	34

6.2	Přístup k databázi.....	35
6.3	Záložní server.....	36
6.4	Konfigurace webového serveru.....	36
7	Závěr.....	37
8	Přílohy bakalářské práce.....	38
	Seznam Literatury.....	39

1 Úvod

Tato bakalářská práce se věnuje problematice portálu vojensko-geografických informací. Jedná se o portál zprostředkovávající přístup ke geografickým informacím a dokumentům různým složkám armády ČR. Zároveň je jeho cílem umožnit geografům v rámci armády sdílet své dokumenty a produkty v elektronické formě přehledným způsobem. Jedná se o webovou aplikaci přístupnou pouze v rámci celoarmádní datové sítě.

Práce je rozdělena do kapitol. Ve druhé kapitole budou objasněny příčiny vzniku portálu a bude stručně popsáno prostředí, ve kterém bude aplikace nasazena. Třetí kapitola se bude věnovat zdůvodnění výběru technologií použitých při vývoji portálu. Čtvrtá kapitola bude zaměřena na analýzu systému. V páté a šesté kapitole budou ukázány zajímavosti z implementace respektive nasazení portálu. V závěru bude nastíněn stav v jakém se systém nachází a budou definovány další potřebné kroky k úspěšnému dokončení dalších vývojových cyklů.

2 Kde se vzala potřeba portálu?

Každý systém vzniká z nějaké příčiny. Touto příčinou je často neuspokojená potřeba nějaké skupiny lidí. V případě informačních systémů je to potřeba získat informace. Pokud jde o geografické informační systémy, týká se tato potřeba geografických informací a cílová skupina jsou většinou geografové. A v případě vojenských geografických informačních systémů je předmětná skupina navíc omezena na vojenské geography. Každé omezení, které jsem v tomto odstavci nastínil, přináší určitá specifika, která jsem při analýze, návrhu a implementaci Portálu Vojensko-Geografických Informací musel brát v úvahu.

V této kapitole popíši potřeby vojenských geografů, které byly příčinou vzniku portálu. Dále nastíním, jak byla situace řešena před vznikem portálu a proč zavedené systémy nemohly dané potřeby uspokojit.

2.1 *Potřeby vojenských geografů*

Když vojenský geograf dostane úkol, musí v první fázi zpracování získat co nejvíce relevantních informací. Aby byly informace skutečně relevantní, musí mít geograf možnost filtrovat je podle různých kritérií. Těmito kritérii jsou:

- typ dokumentu (mapa, geodatabáze, textový dokument, prezentace),
- kategorie mapy (letecký snímek, topografická mapa, fyzicko-geografická mapa ...),
- měřítko mapy,
- území zájmu,
- klíčová slova.

Potřeba první: vyhledávání podle kritérií.

Další úskalí, které musí vojenský geograf překonat, je záplava zkratk. Zkušený geograf se jistě nezalekne zápisu M-33-73-B-c-3 a bude rychle schopen zjistit, k jaké mapě daná nomenklatura patří. Pro méně zkušeného geografa to může být problém. Obě tyto skupiny by určitě ocenily,

kdyby kromě sekvence znaků měly k dispozici také náhled, který informace sděluje bez nutnosti dekodování.

Potřeba druhá: zobrazení náhledů.

Poslední, avšak možná nejdůležitější potřebou, je efektivita. Velké množství zakázek, které vojenští geografové zpracovávají, se opakuje. Vzniká tak velká míra podobnosti, která vybízí k využití již vytvořených podkladů. Geograf pak pouze upraví podklad, aby odpovídal zadání zakázky, a ušetří hodně drahocenného času.

Každý geograf má určitě přehled o svých zakázkách a data pečlivě uschovává pro budoucí využití. Podobná zadání mohou ovšem dostat i různí geografové na různých odděleních.

Potřeba třetí: sdílení za účelem znovupoužití.

2.2 Proč nevyhovují zavedené firemní systémy?

Armáda používá jednotný systém logistického zabezpečení. Všechny předměty logistického zabezpečení, od propisky po tank, lze pomocí tohoto systému objednat. Problém využití tohoto systému je, že nevyhovuje žádné ze tří výše zmíněných potřeb geografů.

Umožňuje sice vyhledávání podle nějakých kritérií, ale tato kritéria nejsou optimalizována pro vyhledávání geografických produktů. Práce se systémem se většinou omezuje na objednávání s využitím znalosti evidenčních čísel, která je nutné získat jinde.

Pokud jde o zobrazení náhledů, systém není schopen zobrazit takové náhledy, které by nesly potřebnou informační hodnotu.

A konečně systém neumožňuje geografům vkládat informace o vlastních produktech. Je tomu tak ze dvou důvodů. První je, že systém je vyvíjen a podporován externí firmou, která má výhradní práva na vkládání nových položek do evidence. Druhým důvodem je, že většina geografů nemá vůbec k systému přístup a objednávky řeší s asistencí útvarového logisty.

Výše zmíněné nedostatky vedly již mého předchůdce k vytvoření webové aplikace, která umožňuje přehledné vyhledávání pomocí nastavování uživatelských filtrů a zobrazování náhledů produktů. Geografům ovšem již neumožňuje vkládání vlastních produktů.

2.3 Analýza stavu

Z počátku se nabízela myšlenka již existující aplikaci pouze rozšířit o nové funkcionality. Při analýze stavu jsem ovšem zjistil, že aplikace je postavena na technologiích, které jen s velkými obtížemi umožňují údržbu a rozšíření. K jejímu běhu byl použit komerční software s uzavřeným kódem a špatnou dokumentací. Aplikace byla implementována v php verze 3, které neumožňuje využití objektově orientovaných postupů, nebo alespoň ne na potřebné úrovni. K aplikaci nebyla vytvořena potřebná programátorská dokumentace.

Na základě vyhodnocení zmíněné analýzy jsem dospěl k závěru, že bude jednodušší a časově méně náročné aplikaci vytvořit znovu s použitím technologií a postupů, které lépe pokryjí naše požadavky a budou snáze udržitelné a rozšiřitelné, než se pokoušet udržovat a vylepšovat aplikaci stávající.

3 Výběr technologií

Při výběru technologií použitých pro vývoj a běh portálu vojensko-geografických informací jsem se snažil, aby jejich použití bylo v daném prostředí co nejschůdnější. Pokud bych volil komerční software, musel bych doufat, že můj požadavek bude schválen, a čekat na nákup softwaru. To by mohlo oddálit nasazení i o několik měsíců. Proto jsem rovnou zvolil open-source, u kterého stačí podat žádost na schválení užívání v resortu.

Abych měl jistotu, že bude portál použitelný pro koncové uživatele, musel jsem vzít v úvahu další omezení:

- Portál je nasazen ve vojenské síti, odkud není přístup do internetu.
- Uživatelé většinou nemají přístup k administrátorskému účtu na svých pracovních počítačích.

Nemohl jsem tedy spoléhat na použití technologií vyžadujících instalaci jakýchkoli doplňků na klientské straně.

3.1 MapServer

Jednou z funkcionalit portálu je zobrazení jednoduché mapky, jejímž účelem je umožnit uživatelům vybrat území zájmu a zobrazit pokrytí požadovanými podklady. Ve staré verzi portálu byl použit komerční produkt Demis web map server. Jelikož se tento produkt neosvědčil, zvolil jsem open-source MapServer. V tomto případě to byla jednoduchá volba, jelikož ho na oddělení již používáme v jiných aplikacích. Mohl jsem tedy využít zkušeností svých spolupracovníků s instalací a konfigurací a v případě potřeby se k nim obrátit pro radu.

MapServer má také dobře zdokumentovanou návaznost na některé další použité technologie a širokou základnu uživatelů. Lze tedy snadno najít řešení konkrétního problému, který se může při implementaci vyskytnout.

Informace k MapServeru i jeho doplňkům jsem čerpal z [4].

3.2 PostgreSQL

Volba DBMS byla ovlivněna hlavně návazností na MapServer. K PostgreSQL existuje rozšíření PostGIS. Je to sada funkcí a operátorů, které umožňují nad daty provádět geografické operace s využitím jejich geometrie. Existuje také doplněk, který usnadňuje import dat ze shapefiles.

V rámci přípravy jsem se účastnil dvou kurzů správy PostgreSQL, kde jsem získal znalosti potřebné k instalaci, konfiguraci a administraci tohoto DBMS.

Mnoho cenných informací o konfiguraci a administraci tohoto DBMS jsem získal z [5]. Informace o rozšíření PostGIS jsem získal z [6].

3.3 PHP5

Jelikož je Portál vojensko-geografických informací webová aplikace a rozhodl jsem se použít volně dostupné technologie, bylo nejschůdnější cestou pro skriptování na straně serveru právě php. Dalším důvodem jsou volně dostupné moduly pro spojení s DBMS PostgreSQL a phpMapScript, s jehož použitím je možné dynamicky přidávat a upravovat vrstvy zobrazované MapServerem.

Předchozí verze portálu byla implementována v php3. Php5 jsem zvolil kvůli možnosti využití objektově-orientovaných postupů, které přináší větší přehlednost kódu a jeho znovupoužitelnost, snazší údržbu a rozšiřitelnost.

Použití starší verze php bylo rovněž nevhodné kvůli možnosti ztráty funkčnosti aplikace při aktualizaci softwaru na serveru.

Když jsem začínal pracovat na Portálu vojensko-geografických informací, teprve jsem se s jazykem php seznamoval. Cestu mi usnadňovalo množství dostupné literatury. Hlavním zdrojem mi byla kniha [3].

Jako bonus se jeví podpora php5 v mém oblíbeném IDE NetBeans.

3.4 GNU/Linux a Apache

Součástí řešení bylo rovněž nasazení portálu. Pro provoz portálu jsem dostal k dispozici server Dell PowerEdge 6300. Jako operační systém jsem zvolil GNU/Linux Ubuntu server 9.10. Tato volba byla podmíněna požadavkem na open-source operační systém a dále ovlivněna především mojí osobní preferencí. Ubuntu používám i na svém osobním počítači, kde jsem podstatnou část

portálu implementoval a testoval. Volil jsem tedy co nejpodobnější systém i na server, abych si usnadnil přenášení nastavení a konfiguračních souborů použitého software.

Webový server Apache jsem zvolil, kvůli jeho snadné konfiguraci a stabilitě při provozu. Je to nejpoužívanější webový server a má tedy širokou základnu uživatelů, z toho plyne snadná dostupnost informací o konfiguraci a řešení případných problémů.

Jako zdroj informací k Ubuntu mi posloužil web [7]. Informace k serveru Apache jsem čerpal převážně z [8].

3.5 Javascript a AJAX

Jak jsem již naznačil v úvodu kapitoly, nemohl jsem se spoléhat na možnost instalace doplňků na klientské stanice. Pro tvorbu prezentační vrstvy a skriptování na straně klienta jsem tedy volil jen technologie, které jsou nativně podporovány všemi běžnými internetovými prohlížeči.

Javascript jsem použil především k obsluze přehledové mapky a AJAX mi umožnil nahradit použití rámců v HTML.

Informace k javascriptu jsem čerpal z webu [9]. Jako zdroj informací k AJAX mi posloužila diskuse na stejném webu.

3.6 Co a proč jsem nepoužil

3.6.1 OpenLayers

Během analýzy trhu jsem narazil na často používaný doplněk openLayers. Je to v javascriptu napsaná knihovna, která umožňuje snadno do webových stránek vložit dynamickou mapku. Rozhodl jsem se openLayers nepoužít a vyžadovanou funkcionalitu implementovat vlastnoručně. Hlavním důvodem pro toto rozhodnutí bylo, že jsem chtěl mít větší kontrolu nad výběrem území, abych mohl provádět validaci extentu před zpracováním SQL dotazu nad tabulkami s daty. Na obsluhu mapserveru stačilo relativně málo kódu a nebylo zapotřebí nasazovat tak robustní nástroj jako je openLayers.

3.6.2 jQuery

Uvažoval jsem o použití frameworku jQuery, ale tuto myšlenku jsem opustil. Vzhledem k

malému množství použitého javascriptu by se nevyplatilo věnovat čas studiu dokumentace tohoto frameworku. Při tvorbě prezentační vrstvy jsem volil spíše strohý design, takže bych stejně nevyužil možností, které jQuery nabízí.

3.6.3 Další nepoužité technologie

Během konzultací se zadavatelem jsme uvažovali i další alternativy. Například použití flash, silverlight, nebo java appletů. Tyto nápady většinou ztroskotaly kvůli předpokladu, že uživatelé si nebudou moci doinstalovat potřebné doplňky. Také jsem zvažoval použití nějakého frameworku pro php, ale opět jsem vyhodnotil čas strávený výběrem frameworku a studiem dokumentace jako příliš cenný a začal jsem se radši věnovat návrhu a implementaci.

4 Analýza systému

Cílem analýzy systému je sestavit konceptuální model systému, na kterém je možné stavět v dalších částech vývojového procesu. Tento model se skládá z řady diagramů a dalších specifikací statických i dynamických vlastností systému. Během vytváření konceptuálního modelu mi byla oporou kniha UML2 : a unifikovaný proces vývoje aplikací [2].

4.1 Požadavky

Před zahájením vývoje jakékoli aplikace je potřeba dobře promyslet různé aspekty, které vedou k úspěšnému dokončení a nasazení. Nejdůležitějším východiskem pro hodnocení aplikace je uspokojení zadavatele, proto je přesná a důkladná specifikace požadavků na software možná nejdůležitější částí vývoje.

Mou výhodou je, že můj zadavatel je můj přímý nadřízený a prostředí, ve kterém je aplikace nasazena mi není cizí.

Spolu se zadavatelem jsme dospěli k požadavkům, které jsou sepsány v následujících podkapitolách. Specifikace je sepsána podle [1].

4.1.1 Hlavní představa

Portál vojensko-geografických informací(dále jen PVGI) je intranetová aplikace umožňující uživatelům vyhledat, zobrazit a stahovat informace a data čtyř různých kategorií (skenované mapy, digitální publikace, produkty PGZ a volně šiřitelné mapky stažené z internetu). PVGI umožňuje geografům vkládat do systému své produkty (produkty PGZ) a sdílet je tak s ostatními geografy. Supervisor je osoba odpovědná za práci jemu přidělených geografů a PVGI mu umožňuje kontrolovat a schvalovat nebo odmítat produkty vložené jemu přidělenými geografy.

4.1.2 Funkční požadavky

1. PVGI bude umožňovat uživatelům vyhledat produkty pomocí procházení stromové struktury.
2. PVGI bude umožňovat uživatelům vyhledat produkty pomocí nastavení omezení (měřítko,

rozsah území, typ dokumentu...)

3. PVGI bude umožňovat uživatelům vyhledávat produkty pomocí kombinace procházení stromové struktury a nastavování omezení.
4. PVGI bude umožňovat uživatelům exportovat seznamy nalezených produktů.
5. PVGI bude uživatelům zobrazovat ilustrační náhledy produktů, pokud budou dostupné.
6. PVGI bude uživatelům umožňovat stáhnout informační náhled produktu, pokud bude dostupný ke stažení.
7. PVGI bude uživatelům umožňovat stáhnout produkt, pokud bude dostupný ke stažení. Alternativně bude PVGI informovat uživatele o možnostech objednání/získání produktu.
8. PVGI bude zabezpečovat neregistrovaným uživatelům přístup pouze k volně dostupným produktům.
9. PVGI bude zabezpečovat registrovaným uživatelům přístup k produktům podle jejich oprávnění.
10. PVGI bude umožňovat geografům nahrávat do systému vlastní produkty.
11. PVGI bude umožňovat geografům vkládat informace o jejich vlastních produktech.
12. PVGI bude umožňovat geografům editovat informace o jejich vlastních produktech.
13. PVGI bude umožňovat geografům mazat informace o jejich vlastních produktech.
14. PVGI bude umožňovat supervisorům komentovat produkty vložené geography, kteří jsou mu přiděleni.
15. PVGI bude umožňovat supervisorům schvalovat produkty vložené geography, kteří jsou mu přiděleni.
16. PVGI bude umožňovat supervisorům odmítat produkty vložené geography, kteří jsou mu přiděleni.
17. PVGI bude umožňovat přístup k doposud neschváleným produktům pouze v rámci oddělení geografa, který produkt vložil.
18. PVGI bude informovat supervisora o každé změně (skupině změn), kterou v systému provedl jemu přidělený geograf.

19. PVGI bude informovat geography o změnách stavu jejich produktů provedených přiděleným supervisorem.
20. PVGI bude umožňovat vložení pouze těch produktů, které budou splňovat vyžadované konvence(vyplnění povinných atributů, georeference ...).

Funkční požadavek	Verze 1	Verze 2	Verze 3
1.	Plně implementováno		
2.	Plně implementováno		
3.	Plně implementováno		
4.		Plně implementováno	
5.	Plně implementováno		
6.	Plně implementováno		
7.	Plně implementováno		
8.	Plně implementováno		
9.	Plně implementováno		
10.		Částečně implementováno	Plně implementováno
11.		Částečně implementováno	Plně implementováno
12.		Částečně implementováno	Plně implementováno
13.		Částečně implementováno	Plně implementováno
14.		Částečně implementováno	Plně implementováno
15.		Částečně implementováno	Plně implementováno
16.		Částečně implementováno	Plně implementováno
17.		Částečně implementováno	Plně implementováno
18.		Částečně implementováno	Plně implementováno
19.		Částečně implementováno	Plně implementováno
20.		Plně implementováno	Plně implementováno

Tabulka 1: Přehled implementace požadavků v jednotlivých verzích aplikace

4.1.3 Nefunkční požadavky

1. PVGI bude dostupný nepřetržitě.

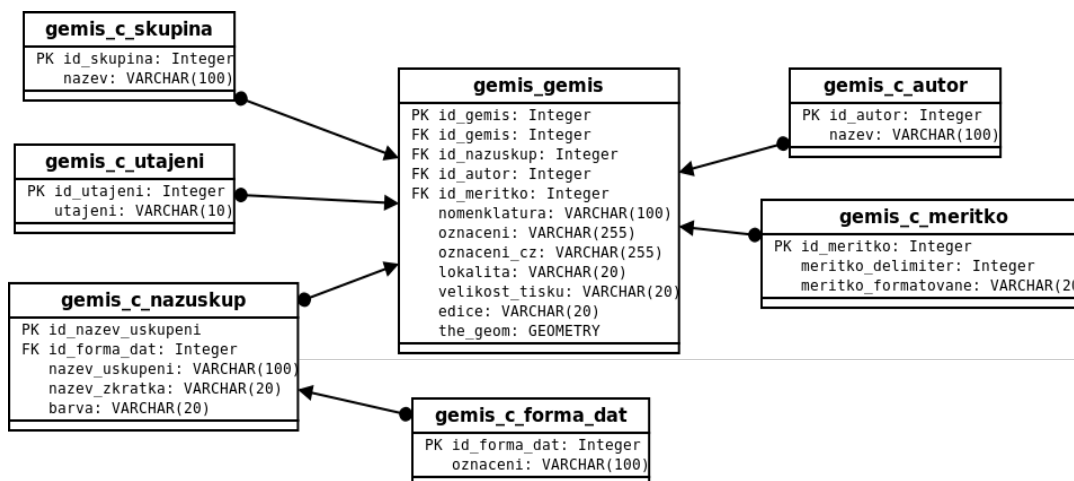
2. PVGI bude dostupný pouze z CADS(celoarmádní datová síť)
3. Pro vývoj a provoz PVGI bude využit jen software šířený pod GNU/GPL nebo jinou svobodnou licencí.
4. PVGI bude instalováno na stávajícím hardwaru OdGP Praha. Nepočítá se s nákupem nového hardwaru, pokud nedojde ke kritickým problémům.
5. PVGI bude zařazeno do automatizovaného systému záloh OdGP Praha.

4.2 Datová analýza

Datové zdroje, ze kterých portál čerpá, se dají seskupit do tří logických skupin. Každá z těchto skupin má svá specifika. Mou snahou bylo připravit aplikaci tak, aby bylo možné přistupovat k různým datům jednotným způsobem.

4.2.1 GEMIS

První skupinou je databáze GEMIS vznikající mimo oddělení geografické podpory. Tato databáze je primárně určena pro jiné aplikace, a proto je z hlediska portálu vojensko-geografických informací přístupná jen pro čtení.



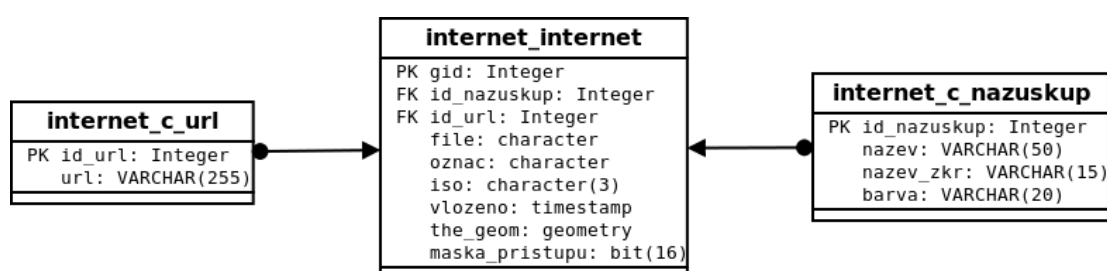
Obr. 1: Konceptuální schéma databáze GEMIS

Na obrázku Obr. 1 je znázorněna struktura databáze GEMIS třídícím UML diagramem. Označení PK u atributů znamená, že se jedná o primární klíč. Označení FK znamená, že atribut je cizí klíč. Je patrné, že všechny tabulky mají prefix gemis oddělený podtržítkem. Tím je označena

příslušnost tabulky k dané databázi. Číselníky mají navíc prefix c, který je také oddělen podtržítkem. Stejná konvence je použita u všech konceptuálních schémat v této práci.

4.2.2 Internet

Druhým zdrojem dat je databáze internetových map. Jedná se o mapy, které jsou volně dostupné na internetu a jsou volně šiřitelné. Jedná se spíše o schematické mapky, které většinou nemají udané měřítko ani přesnou georeferenci. Tyto mapky jsou v pravidelných intervalech stahovány z internetu a vkládány do portálu vojensko-geografických informací, aby byly dostupné v armádním intranetu a bylo je možné využít při zpracování dalších dokumentů.

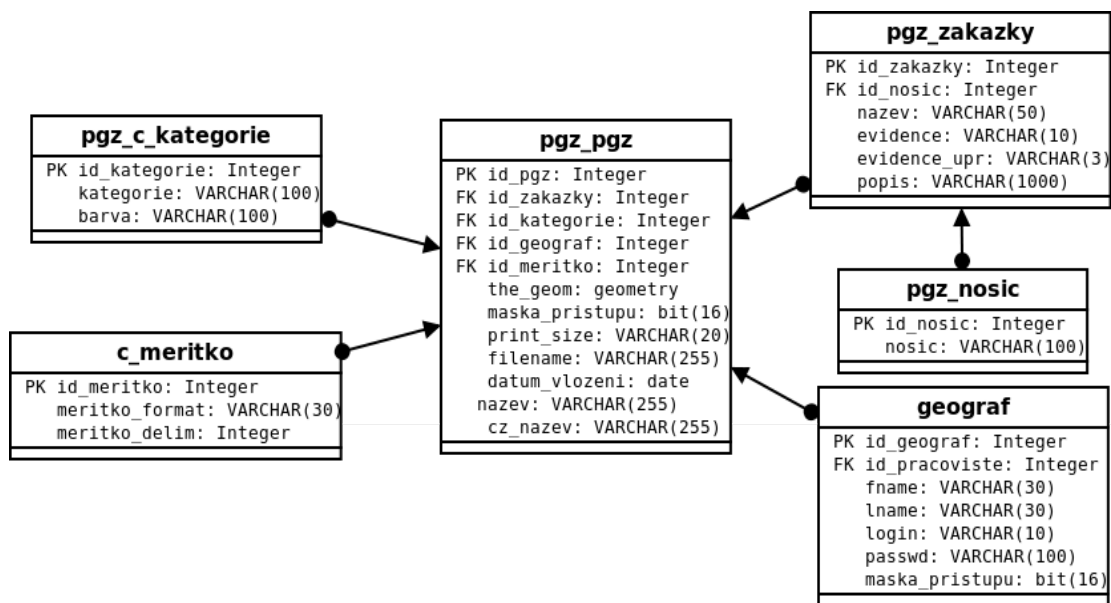


Obr. 2: Konceptuální schéma databáze internetových map

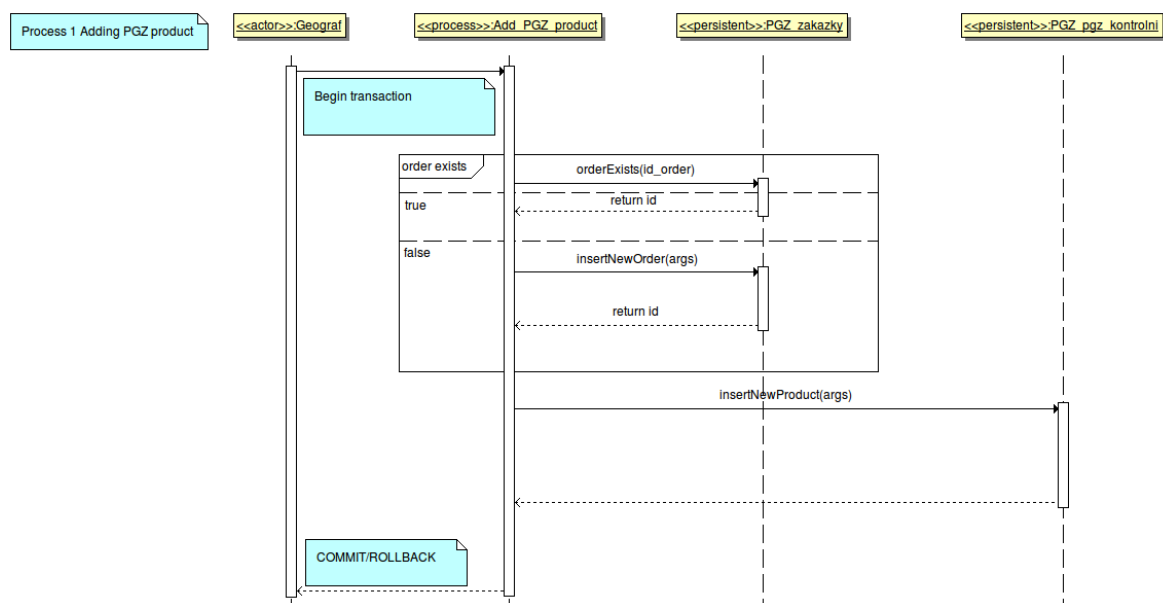
Do této databáze se vkládají a aktualizují data v pravidelných intervalech a dělá to vždy jen administrátor databáze, proto není potřeba blíže analyzovat procesy a konceptuální schéma je z hlediska portálu vojensko-geografických informací postačující.

4.2.3 PGZ

Posledním zdrojem dat je databáze přímého geografického zabezpečení. Tato databáze shromažďuje informace o produktech vyrobených na odděleních geografické podpory a dále produkty geografů zařazených u vojsk. Struktura této databáze je již složitější a vyskytují se u ní složitější procesy. Jelikož konceptuální schéma nemůže postihnout dynamické aspekty aplikace, bylo potřeba přidat popis složitějších procesů sekvenčními diagramy na obrázcích Obr. 4 a Obr. 6.



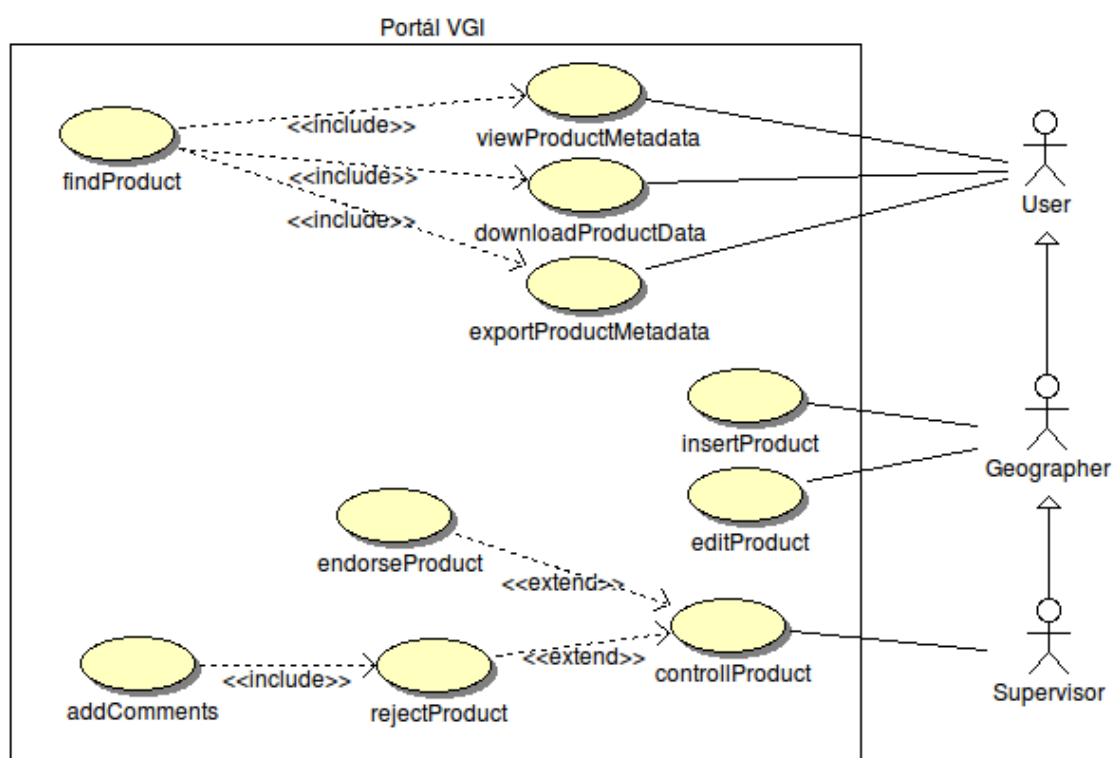
Obr. 3: Konceptuální schéma databáze produktů přímého geografického zabezpečení



Obr. 4: Sekvenční diagram procesu vložení produktu PGZ

4.3 Případy užití

V další fázi analýzy jsem vytvořil diagram případů užití, který mi pomohl identifikovat aktéry a funkce, které od systému očekávají.



Obr. 5: Diagram případů užití

Z diagramu je patrné, že k systému budou přistupovat tři typy aktérů.

4.3.1 User

Tento typ aktéra má přístup k systému jen pro čtení. Může v systému vyhledat požadovaná data, zobrazit si jejich náhledy a v případě jejich dostupnosti je i stáhnout. Nemůže vkládat vlastní produkty. Je to nejobecnější typ aktéra, který k systému přistupuje. Ostatní typy aktérů jsou od typu User odvozeni.

4.3.2 Geographer

Geographer je specializací aktéra User. Dědí všechny jeho vlastnosti a přidává další. Geographer je aktér, který může do systému vkládat vlastní produkty a dokud nejsou schváleny má také možnost je editovat.

4.3.3 Supervisor

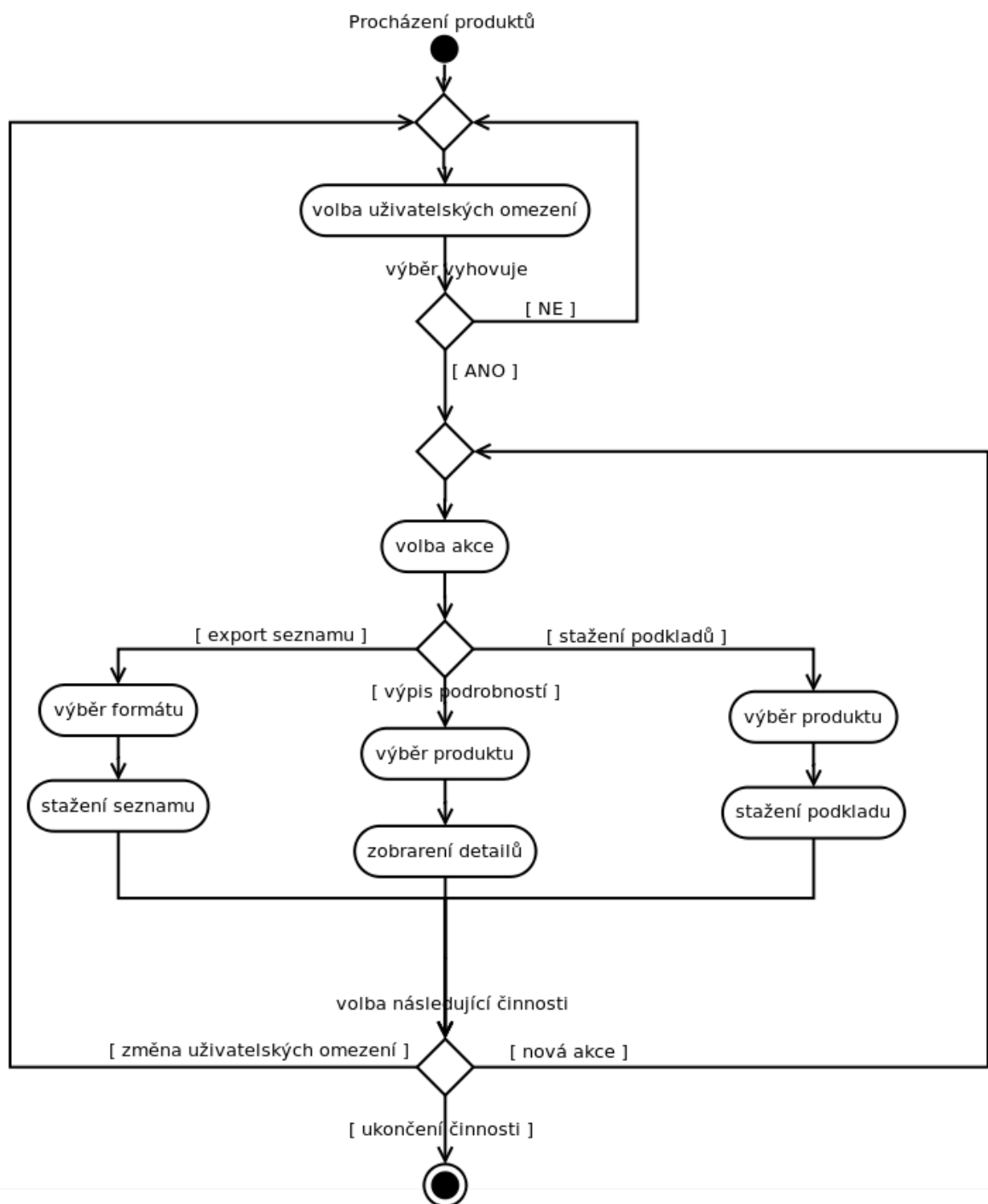
Posledním typem aktéra je Supervisor. Supervisor je specializací aktéra Geographer. Může tedy do systému vkládat vlastní produkty. Dále je tento aktér zodpovědný za kontrolu produktů vložených aktéry typu Geographer. Každý Supervisor má přidělenou množinu aktérů typu Geographer, jejichž produkty kontroluje.

4.4 Diagramy aktivit

K popisu chování systému jsem využil diagramy aktivit. Tyto diagramy se využívají k modelování složitých činností, které se rozdělí na části a jejich vazby. Je tak možné činnost snadněji pochopit a implementovat.

4.4.1 Procházení produktů

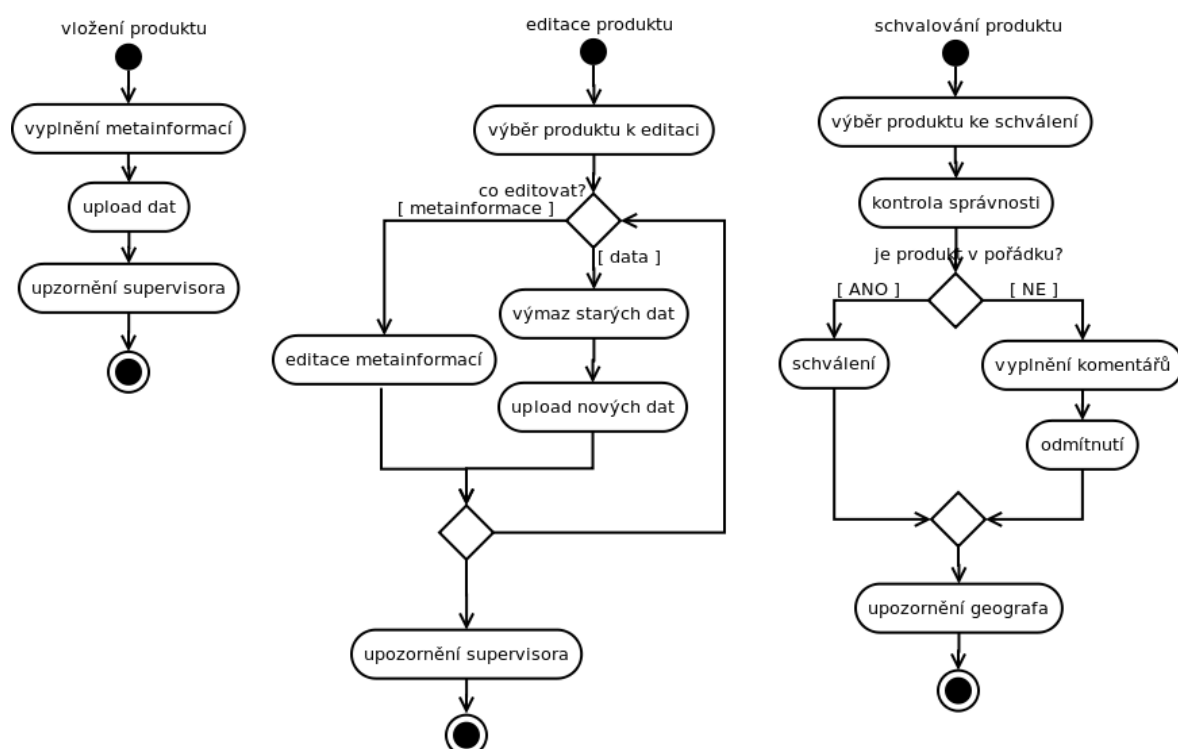
Nejčastěji prováděná činnost všech uživatelů portálu je procházení produktů. Ať už chce uživatel produkt stáhnout nebo ověřit jeho skladovou dostupnost v papírové formě nebo potřebuje přehled pokrytí zájmového prostoru mapovými podklady, vždy potřebuje projít produkty, aby získal požadované informace. Tuto činnost popisuje diagram na obrázku Obr. 6.



Obr. 6: Diagram aktivity procházení produktů

4.4.2 Přidání nového produktu

Přidání nového produktu do systému probíhá ve dvou krocích. V prvním kroku uživatel v roli geografa vloží nový produkt do systému a vyplní k němu potřebné metainformace. Ve druhém kroku uživatel v roli supervisora zkontroluje správnost metainformací a vložených dat a má možnost produkt schválit nebo odmítnout. Pokud je produkt odmítnut, geograf má možnost editovat vložené metainformace a nahrát nová data na základě komentářů, které supervisor při kontrole uvedl. Tyto činnosti popisují následující diagramy.

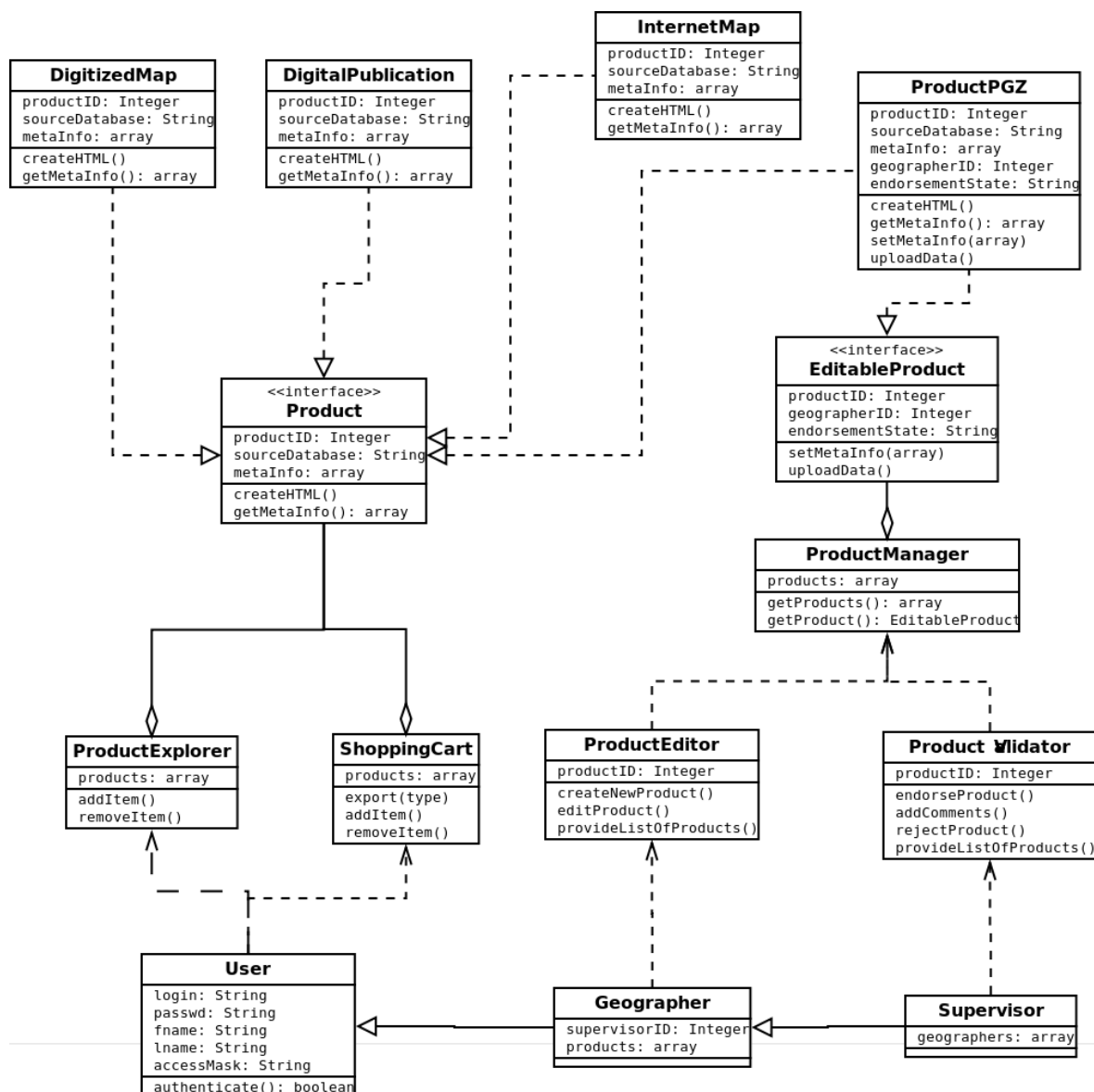


Obr. 7: Diagramy aktivit popisující vkládání nových produktů

4.5 Třídní diagram

V poslední části analýzy jsem vytvořil analytický třídní diagram, který obsahuje analytické třídy a jejich relace. Spolu s ostatními diagramy a specifikacemi tvoří konceptuální model systému. Analytický třídní diagram neobsahuje všechny třídy, které se vyskytnou v implementaci, a třídy v něm obsažené nejsou kompletně specifikovány. Analytický třídní diagram je částí konceptuálního

modelu a v dalších vývojových cyklech se bude zpřesňovat a dále doplňovat.



Obr. 8: Analytický třídní diagram

4.5.1 Generalizace aktéra

Generalizace aktéra byla již naznačena v diagramu případů užití. Aktéři od sebe dědí všechny atributy a s každou úrovní specializace přidávají nové. Je zřejmé, že uživatel v roli *User* má přístup pouze k procházení dat a část systému, která se stará o vkládání nových produktů, mu není

přístupná.

4.5.2 Rozhraní

V třídním diagramu na obrázku Obr. 8 jsou zaneseny čtyři třídy produktů. Aby bylo možné k nim přistupovat jednotným způsobem, všechny implementují rozhraní *Product*. Toto rozhraní definuje proměnné a metody, které musí každá z tříd obsahovat. Třída *ProductPGZ* navíc implementuje rozhraní *EditableProduct*, které zaručuje, že třída obsahuje proměnné a metody, které umožní vkládání a editaci nových produktů do systému.

4.5.3 Kolekce

V systému jsou tři třídy, jejichž úlohou je udržovat seznamy produktů a zprostředkovávat přístup k jednotlivým položkám těchto seznamů.

Třída *ProductExplorer* slouží k procházení produktů. Kdykoli uživatel změní nastavení uživatelských filtrů, změní se obsah pole uvnitř instance této třídy tak, aby odpovídal výsledkům nalezeným na základě uživatelských požadavků.

Třída *ShoppingCart* udržuje seznam produktů, které uživatel označil jako vybrané, a implementuje metody, které umožňují export tohoto seznamu v různých formátech.

Třída *ProductManager* slouží aktérům typu *Geographer* a *Supervisor* při vkládání a kontrole nových produktů. Pro aktéra typu *Geographer* obsahuje seznam jeho zatím neschválených produktů a pro aktéra typu *Supervisor* seznam produktů, které čekají na schválení.

5 Implementace

5.1 Masky přístupu

Jedním z požadavků na portál vojensko-geografických informací je, aby umožnil registrovaným uživatelům přístup k produktům podle jejich oprávnění. Původně byl problém řešen pomocí úrovní oprávnění přiřazených každému uživateli a každému produktu. Každý uživatel měl tedy přístup k produktům s úrovní oprávnění nižší nebo rovnou úrovni oprávnění daného uživatele. Toto řešení se ovšem ukázalo jako málo pružné. Vyskytly se totiž případy, kdy uživatel s nízkou úrovní oprávnění nemohl přistoupit k produktu, na kterém spolupracoval, protože produktu byla při vkládání nastavena úroveň vyšší než jemu dostupná.

Bylo tedy potřeba vyvinout systém práv takový, aby bylo možno přidělovat práva skupinám uživatelů, ale také byla zachována možnost selektivního nastavování práv každého uživatele.

Navrhl jsem systém přidělování a kontroly přístupových práv na základě binární masky. Šestnáctibitová binární maska je teď přidělena každému uživateli, každému produktu a navíc každé skupině uživatelů. Skupiny byly stanoveny na základě organizační struktury armádních geografů a dalších logických vazeb.

Název skupiny	Přidělená maska
A	0000000000000001
B	0000000000000010
C	0000000000000100
D	0000000000001000

Tabulka 2: Příklad rozdělení do skupin

Produktům se maska stanovuje podle toho, které skupiny uživatelů mají mít k danému produktu přístup. Pokud například má být produkt dostupný členům skupin A a B, ale nikomu jinému, bude mít nastaveny odpovídající bity masky na '1'. Výsledná maska takového produktu by tedy byla '0000000000000011'. Při ověřování práv na přístup k danému produktu se aplikuje logický součin. Aby byl přístup k produktu udělen, musí tedy mít obě ověřované masky alespoň na jedné pozici shodnou hodnotu '1'.

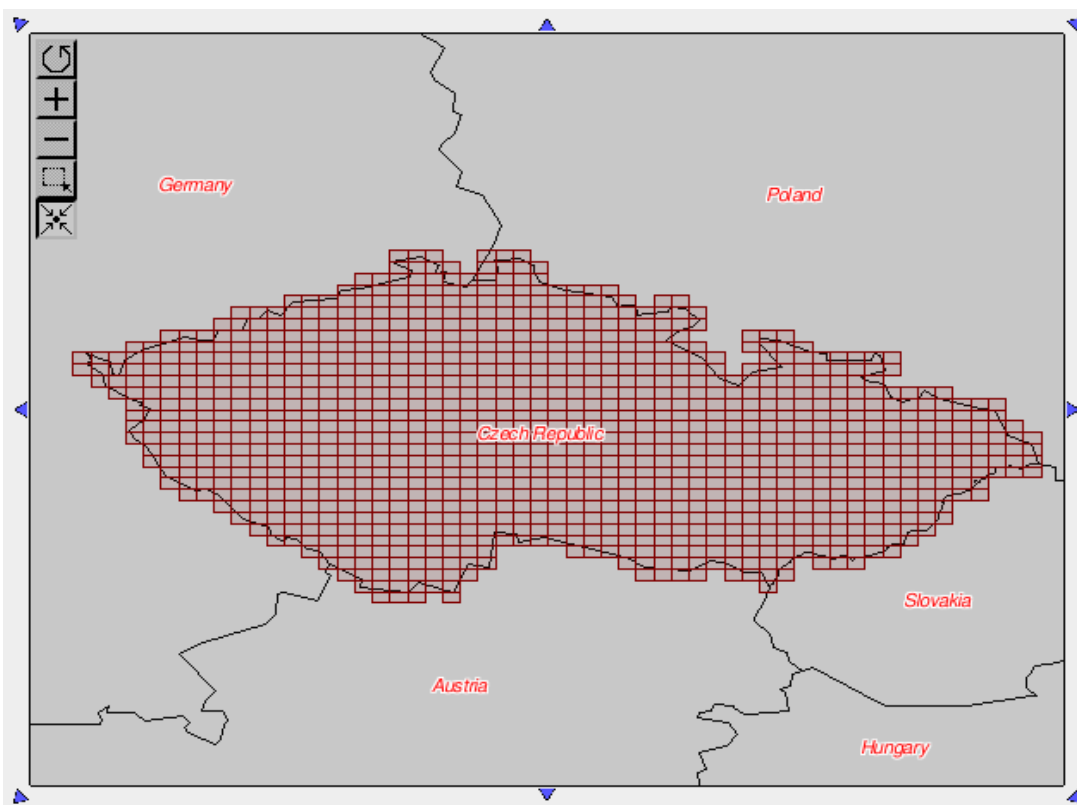
Základní maska každého uživatele je '0000000000000000'. Maska, která se při ověřování přístupu k produktu testuje, je získána logickým součtem masky uživatele a masky skupiny, do které je daný uživatel zařazen. V případě, že se administrátor systému rozhodne, že by nějaký uživatel ze skupiny C měl mít mimo svou skupinu přístup také k produktům s oprávněním pro skupiny A a B, stačí, když jeho základní masku '0000000000000000' upraví na '0000000000000011'. Když potom tento uživatel žádá přístup k nějakému produktu, ověřovaná maska vypadá takto '0000000000000011' a má tedy přístup k produktům skupin A,B a C.

Je tedy možné přidělit práva celé skupině tak, aby všichni její členové měli práva přístupu ke všem jejím produktům a zároveň, pokud je to potřeba, tyto produkty skrýt před skupinou jinou. V případě potřeby je možné nastavit privilegovanému uživateli přístup k produktům i mimo jeho skupinu.

Po přihlášení uživatele do systému se vypočítá jeho maska a ta je uložena do superglobálního pole `$_SESSION`. Později, když je potřeba ověřit přístup k produktu, se tato maska vloží do šablony SQL dotazu a ověří se na úrovni databáze.

5.2 Počítání souřadnic

Jedním z hlavních prvků grafického uživatelského rozhraní je přehledová mapa. Tato mapa slouží k výběru zájmového území a zobrazení pokrytí daného území vybranými produkty. Ukázka přehledové mapky je na obrázku Obr. 9.



Obr. 9: Ukázka přehledové mapky s pokrytím TM25

Na obrázku Obr. 9 jsou v levém horním rohu vidět ovládací tlačítka. První slouží k obnovení mapky, druhé k přiblížení, třetí k oddálení, čtvrté k obdélníkovému výběru a poslední k centrování na vybraný bod. K obsluze všech těchto tlačítek bylo potřeba využít skriptování na straně klienta. V mém případě tedy javascript. Aby tlačítka, jejichž funkcí je změna vybraného území, mohla fungovat správně, bylo potřeba odchyťávat souřadnice kliknutí myši nad obrázkem. Tyto souřadnice bylo následně potřeba přepočítat ze souřadnic v pixelech na souřadnice zeměpisné.

Při řešení tohoto problému jsem narazil na dvě překážky.

1. Uživatel má možnost změnit velikost přehledové mapky.
2. Vždy je potřeba znát aktuální zeměpisné souřadnice rohů zobrazené mapky.

Navrhl jsem dvě funkce, které tento přepočet řeší.

```
/**
```

```
 * funkce na prepocet souradnic v pixelech na geograficke
```

```

*/
function countGeograficalCoordinateX(pixelCoordinate) {
    var geoCoordinateX = 0;
    var map_width = document.getElementById("id_map_width").value;
    var geo_minx = document.getElementById("geo_minx").value;
    var geo_maxx = document.getElementById("geo_maxx").value;
    x = ((geo_maxx - geo_minx) * pixelCoordinate) / (map_width);
    geoCoordinateX = parseFloat(geo_minx) + x; //jinak scita string
    return geoCoordinateX;
}

function countGeograficalCoordinateY(pixelCoordinate) {
    var geoCoordinateY = 0;
    var map_height = document.getElementById("id_map_width").value * (3/4);
    var geo_miny = document.getElementById("geo_miny").value;
    var geo_maxy = document.getElementById("geo_maxy").value;
    y = ((geo_maxy - geo_miny) * pixelCoordinate) / (map_height);
    geoCoordinateY = geo_maxy - y;
    return geoCoordinateY;
}

```

Obě překážky jsem překonal obdobným způsobem. Hodnoty, které při výpočtech používám, uchovávám ve skrytých formulářových polích HTML dokumentu. Při změně velikosti mapky nebo změně vybraného území je po obnovení dokumentu v těchto formulářových polích opět uložena aktuální velikost mapky a aktuální zeměpisné souřadnice jejích rohů.

5.3 Zobrazení pokrytí

K zobrazení pokrytí zájmového území vybranými produkty jsem využil rozšíření phpmapscript, které umožňuje dynamické přidání vrstvy do předem vytvořeného objektu mapy. Ve skutečnosti bylo potřeba pro každý typ produktu vytvořit dvě vrstvy. První slouží k zobrazení obrysu a druhá k vykreslení výplně s desetiprocentní opacitou. Pro tento účel jsem vytvořil v php funkci paintLayer, které předávám argumenty groupID, color, query. Argument groupID slouží pro jednoznačné pojmenování vrstvy. Argument color nese barvu, jakou se má vrstva vykreslovat, a argument query je SQL dotaz, který umožní mapserveru získat potřebnou geometrii.

```

function paintLayer($groupID, $color, $query) {
    $connection = "user=xxxxxx password=***** dbname=web_db
host=10.192.255.34 port=5432";

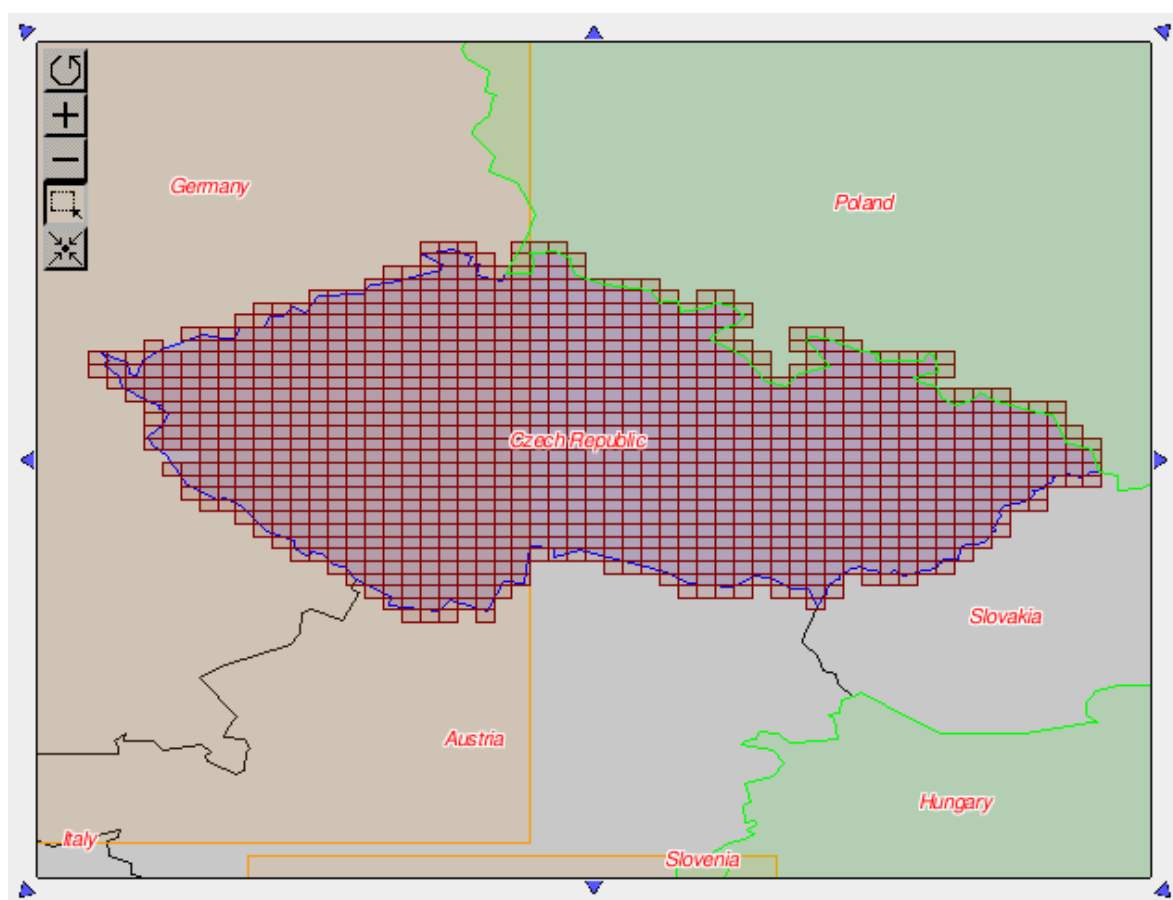
```

```

        $outliningLayer = ms_newLayerObj($this->map);
        $transparentLayer = ms_newLayerObj($this->map);
        $outliningLayer->set("name", "pvgi_layer_.$groupId._outline");
        $transparentLayer->set("name", "pvgi_layer_.$groupId._transparent");
        $outliningLayer->setConnectionType(MS_POSTGIS);
        $transparentLayer->setConnectionType(MS_POSTGIS);
        $outliningLayer->set("connection", "$connection");
        $transparentLayer->set("connection", "$connection");
        $outliningLayer->set("type", MS_LAYER_POLYGON);
        $transparentLayer->set("type", MS_LAYER_POLYGON);
        $outliningLayer->set("data", "$query");
        $transparentLayer->set("data", "$query");
        $outliningLayer->set("status", MS_ON);
        $transparentLayer->set("status", MS_ON);
        $outliningLayer->updateFromString("LAYER CLASS STYLE COLOR -1 -1 -1
        OUTLINECOLOR ".$color." END END END");
        $transparentLayer->updateFromString("LAYER OPACITY 10 CLASS STYLE
        COLOR ".$color." END END END");
    }

```

Na obrázku Obr. 10 je zobrazen výsledek po přidání čtyř dynamických vrstev.



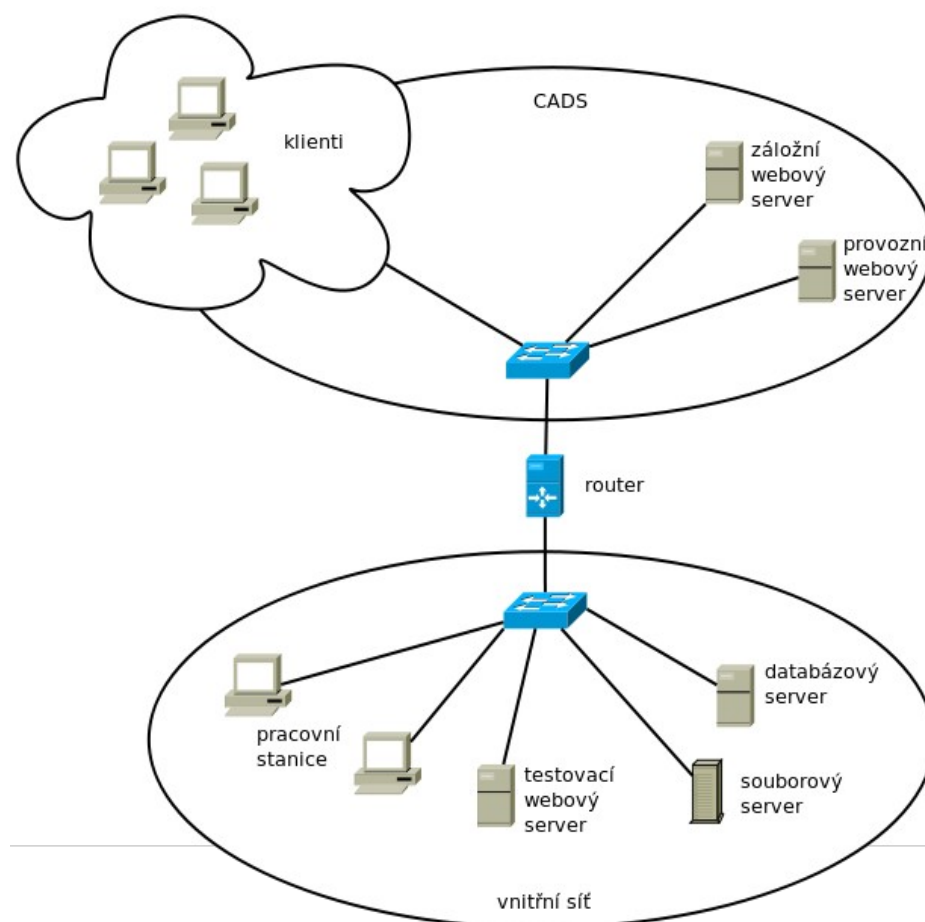
Obr. 10: Přehledová mapka - dynamicky přidané vrstvy

6 Nasazení

Vývoj a provoz aplikace obvykle závisí na více IT pracovnících. Při vývoji se uplatňují konsultanti, analytici, návrháři a programátoři. Při nasazení a provozu spolupracují s programátory administrátoři a někdy také síťáři. Pokud jde o portál vojensko-geografických informací zastávám všechny tyto role alespoň částečně já. Jistou nevýhodou tohoto způsobu organizace je množství práce, která leží na mých bedrech. Výhodou na druhou stranu je, že takto mohu získat široký přehled znalostí z celé IT oblasti a účastním se celého životního cyklu aplikace ve všech jeho částech. V této kapitole ve stručnosti představím činnosti související s nasazením portálu vojensko-geografických informací do celoarmádní datové sítě.

6.1 Topologie

Portál vojensko-geografických informací je webová aplikace a jako taková je z definice typu klient-server. V dnešní době je většina webových aplikací postavena na třívrstvé (nebo vícevrstvé) architektuře. Portál vojensko-geografických není výjimkou.



Obr. 11: Zjednodušená topologie sítě

Na obrázku Obr. 11 je zachycena zjednodušená topologie sítě, kde je portál nasazen. Prezentační vrstva je zastoupena klientskými stanicemi v oblaku v CADS (celoarmádní datová síť), aplikační vrstva je reprezentována webovými servery a datovou vrstvu tvoří databázový a souborový server ve vnitřní síti oddělení geografické podpory.

6.2 Přístup k databázi

Z topologie na obrázku Obr. 11 je patrné, že databázový server není přímo přístupný serveru webovému. Ačkoli stojí oba ve stejné místnosti, jsou patrně připojeni na úplně jiné síťové prvky. Mé administrátorské pravomoci zahrnují počítače ve vnitřní síti, router a webové servery v CADS. Aktivní prvky a ostatní počítače mimo vnitřní síť jsou mimo moji pravomoc. Prostředí mezi webovými servery a routerem je pro mě neznámé, a proto potenciálně nebezpečné.

Na routeru jsem nastavil směrování tak, aby požadavky na port 5432/tcp byly směrovány do

vnitřní síť na databázový server. Zároveň jsem povolil přístup na port 5432 jen z adres záložního a provozního webového serveru.

6.3 Záložní server

V současné době probíhá testovací provoz mé aplikace. To co jsem na obrázku Obr. 11 nazval záložním serverem je ve skutečnosti webový server, kde stále běží předchozí verze portálu vojensko-geografických informací. Až bude nová verze odladěna a nasazena do ostrého provozu měl by se tento server podle plánu opravdu stát záložním.

6.4 Konfigurace webového serveru

Jak již bylo zmíněné v kapitole 3, portál je nasazen na webovém serveru Apache 2 na operačním systému GNU/Linux Ubuntu server 9.10. Konfigurační soubory webového serveru Apache 2 jsou na operačních systémech GNU/Linux standardně uloženy v adresáři */etc/apache2* a zdrojové soubory aplikací se umísťují do adresáře */var/www*. Pro svou aplikaci jsem vytvořil adresář */var/www/pvgi* a do něj jsem umístil zdrojové soubory. Poté jsem v souboru */etc/apache2/sites_available/default* vytvořil záznam, který zpřístupnil soubory webovému serveru.

```
Alias /portal/ "/var/www/pvgi/"
<Directory "/var/www/pvgi/">
    Options Indexes MultiViews FollowSymLinks
    AllowOverride None
    Order allow,deny
    Allow from all
</Directory>
```

Po restartu webového serveru byla aplikace zpřístupněna z CADS na adrese:
<http://mapserver.topo.acr/portal/>.

7 Závěr

Cílem této bakalářské práce bylo vytvoření portálu vojensko-geografických informací. Během zpracování této práce jsem provedl funkční a datovou analýzu, sestavil jsem konceptuální model systému a implementoval jsem první verzi této webové aplikace. Plné uspokojení požadavků stanovených zadavatelem se očekává až ve verzi třetí. Byly uspokojeny funkční požadavky 1-9 s výjimkou požadavku 4. Stav implementace tedy, podle tabulky požadavků prezentované v kapitole Analýza systému, odpovídá dokončené první verzi.

Dalším krokem směrem k dokončení projektu je nasazení první verze do testovacího provozu a úspěšné otestování. Následně budou spolu se zadavatelem podrobněji specifikovány požadavky na další rozšíření a započnou práce na druhé verzi.

8 Přílohy bakalářské práce

1. Uživatelská příručka
2. Programátorská příručka
3. Disk CD

Seznam Literatury

- [1] WIEGERS, Karl E. *Požadavky na software : Od zadání k architektuře aplikace*. 2. vydání v češtině. Brno : Computer Press a.s., 2008. 444 s. ISBN 978-80-251-1877-1.
- [2] ARLOW, Jim; NEUSTADT, Ila. *UML2 : a unifikovaný proces vývoje aplikací*. 2. vydání. Brno : Computer Press a.s., 2008. 567 s. ISBN 978-80-251-1503-9.
- [3] GUTMANS, Andi; BAKKEN, Stig Saether; RETHANS, Derick. *Mistrovství v PHP 5* . 1. vydání. Brno : Computer Press a.s., 2007. 656 s. ISBN 978-80-251-1519-0.
- [4] *Mapserver.org* [online]. 2009 [cit. 2010-04-17]. MapServer 5.6.1 Documentation. Dostupné z WWW: <<http://mapserver.org/documentation.html>>.
- [5] STĚHULE, Pavel. *PostgreSQL* [online]. 2008 [cit. 2010-04-17]. Dostupné z WWW: <<http://www.postgres.cz/index.php/PostgreSQL>>.
- [6] *PostGIS* [online]. 2001 [cit. 2010-04-17]. Dostupné z WWW: <<http://postgis.refractions.net/documentation/>>.
- [7] *Wiki - ubuntu Česko* [online]. 2008 [cit. 2010-05-03]. Dostupné z WWW: <<http://wiki.ubuntu.cz/>>.
- [8] *The Apache : HTTP server project* [online]. 2009 [cit. 2010-05-03]. Dostupné z WWW: <<http://httpd.apache.org/>>.
- [9] JANOVSKEÝ, Dušan. *Jak psát web : o tvorbě internetových stránek* [online]. 2000 [cit. 2010-05-03]. Dostupné z WWW: <<http://www.jakpsatweb.cz/>>.